(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
4 October 2001 (04.10.2001)

PCT

(10) International Publication Number
WO 01/73533 A1

(51) International Patent Classification⁷: G06F 1/26, 1/28, 1/30, 11/30, 12/14, 15/173, H04L 9/00, 9/32

(21) International Application Number: PCT/US01/09889

(22) International Filing Date: 28 March 2001 (28.03.2001)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
60/192,426 28 March 2000 (28.03.2000) US

(71) Applicant *(for all designated States except US)*: THE WINGARD COMPANY [US/US]; 216 Heatherdown Road, Decatur, GA 30030 (US).

(72) Inventors; and
(75) Inventors/Applicants *(for US only)*: WINGARD, Peter, F. [US/US]; 217 Heatherdown Road, Decatur, GA 30030 (US). SIMMONS, John, C. [US/US]; 764 Walnut Bend, Cordova, TN 38138 (US).

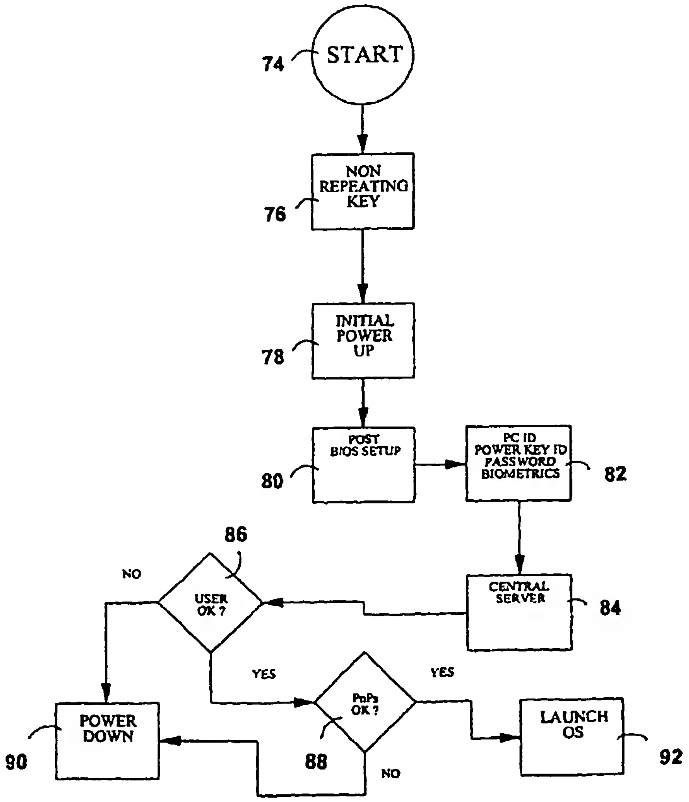
(74) Agents: DOUGHERTY, Ralph, H. et al.; Dougherty & Clements LLP, Suite 400, 6230 Fairview Road, Charlotte, NC 28210 (US).

(81) Designated States *(national)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZW.

(84) Designated States *(regional)*: ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE,

[Continued on next page]

(54) Title: SYSTEM AND METHOD FOR SAFEGUARDING ELECTRONIC FILES AND DIGITAL INFORMATION IN A NETWORK ENVIRONMENT



(57) Abstract: A system for securing electronic files on a network by restricting access to only authorized users, monitoring use of remote work stations and embedding scalable self executing applets within the electronic files which may cause one of a variety of remedial actions when accessed improperly. To access a particular file, an end user must first (74) successfully power up (78) the remote workstation by using a unique code (82) which must be verified by a power supply component (80) . Thereafter, the end user must pass through a series of validation points before accessing the network (84). The network (84) then monitors and grants permission for each secured activity initiated by the user while using the remote workstation on the network. If the user fails to follow predetermined operating practices, the network (84) has the ability to remove and lock out the offending user from the network (84) and thereby remove his access (90) to files protected by the network.

IT, LU, MC, NL, PT, SE, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

Published:

— *with international search report*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

SYSTEM AND METHOD FOR SAFEGUARDING ELECTRONIC FILES AND DIGITAL INFORMATION IN A NETWORK ENVIRONMENT

CROSS-REFERENCE TO RELATED APPLICATION

This application claims the benefit of U.S. Provisional Application No. 60/192,426, filed March 28, 2000.

FIELD OF THE INVENTION

The present invention relates to a method and system for protecting electronic files, digital information or the like from theft and sabotage by an unauthorized user of a remote workstation, PC, which is connected to a secured network, and more particularly the invention relates to a method and system for validating a specific user and of employing the use of scalable self executing applets that are attached to the electronic files and digital information, which, when improperly accessed, destroys the electronic files, corrupts the PC's hard drive, or signals the network of the unauthorized use.

BACKGROUND OF INVENTION

In recent years, the power of business computing networks has moved away from large mainframe computers to more economical and versatile desktop and laptop computer networks. Medium and large businesses have gained computing power, economy, and versatility, but they have lost control and network security. With the rise of computer networks, the risk of information theft and/or sabotage has increased. Such theft and sabotage can be wrought by "hackers" or, more importantly, by employees of a business organization. Notwithstanding the known threat of a "hacker" (an individual who attempts to gain access to a company's data stored on a system computer), employee theft and/or sabotage of secure information presents a very real threat.

By its very definition, the personal computer, or PC, is personal. Once an organization assigns a computing asset, or PC, to an employee, that PC is no longer in the company's control. In essence, it becomes the personal property of that individual. When such an employee seeks to obtain sensitive company information without proper authorization, the company is presented with a security problem as the employee is in possession of the PC. Moreover, when company computing assets are lost or stolen or when proprietary/sensitive information is accessed by unauthorized individuals using independent computing assets, the need to regain control may be crucial to the company's network security. By examining the prior art, it is evident that various systems have been implemented which attempt to provide security to organizations from unauthorized users.

Traditional methods to safeguard electronic files and other information over a network began with physical access control methods, such as access restriction to a building, floor, room, or even a computer by using, for example, a password, a key, or a fingerprint scanner. Most network security procedures have some form of this traditional method incorporated therein. The obvious rationale for implementing these methods is that the more restrictions you place on the access to a secure computer, the better the security. However, the restrictions and control methods implemented come with several disadvantages. For example, the better the security *via* physical access control, the harder the computing asset/network is to use. For yet another example, physical access control methods cannot control what happens outside the physical limits of the network, such as outside the secured building. For yet another example, individuals can simply copy a file to a portable storage diskette or CD and remove it from the facility.

Other art has disclosed methods of employing passwords to prohibit access by unauthorized users. These methods have become the dominant form of protection of information amongst businesses today. However, the use of passwords at access points have proven to be ineffective. Current forms of password access control are susceptible to password-interception. In a typical method of a password transaction, the system administrator (i.e., the organization) inputs a password into the system secured memory. Subsequently, a finite number of users are provided with the password. Then, a user enters the password into an input device (i.e., a keyboard) at a predetermined prompt. Thereafter, the password data is passed through the network where a comparison takes place. A secured memory password which is located in the network server is retrieved and compared with the user password. If there is a match, access is permitted and several functions and various applications are enabled. If there is no match, access is denied.

Interception of a password can occur in multiple manners. By way of example, an individual can simply peer at the user's input strokes to ascertain the password. By way of another example, inception can occur by using certain software applications which are designed for capturing or displaying memory data (e.g., hacking software). By way of a still further example, an individual can connect electronic equipment which may read and interpret bus or port digital traffic. Once interception is achieved, an organizations electronic files or other information are available for theft and/or sabotage.

Still other art has disclosed the use of encryption methods to protect data and other information. In a typical encryption method, when a file is stored in memory it is encrypted. Thereafter, when a user desires to access the file he is prompted for a decryption key. Once the key is provided, the file is decrypted. A disadvantage of encryption is that when a file is decrypted the user has full access to the information, including but not limited to copying the same. A further disadvantage is that this method uses a significant amount of memory resources to function.

Still other art teaches the use of PKI's (public key infrastructure). PKI enables users of a basically unsecure public network to securely exchange data. The method uses a public and private cryptographic key pair. The PKI provides for a digital certificate that can identify an individual and directory services that can store and, when necessary, revoke the certificates. A Certificate Authority (CA) issues both the public and private keys. The private key remains unshared while the public key is distributed. A user with a private key can decrypt text that has been encrypted by another user who has the public key. Similar to the other art cited above, PKI's have several disadvantages. By way of example, a user's private sign-in key is stored on his/her computer. There, it is subject to viruses and other malicious programs. Even if the user's key is in a safe computer, it is not necessarily in a physically controlled environment. In addition, a user has no known way of verifying the certificate of another. Indeed, many CA's issues certificates with a name. Thus, although a user might only know one John Doe, the CA may know numerous. Determining what John Doe is known or intended by the user is virtually impossible.

With all of the above cited art, no provision is made for a continuous security measure after the initial access point. Moreover, if an organization wishes to employ more than one of the above methods, separate applications must be used. Thus, it is evident that there is a general need in the art for a more convenient and secure method of protecting electronic files and digital information from unauthorized access. It is also evident that there is a need to provide a constant measure of security for the duration of the user's access, not just at the access point. Finally, it is evident that there is a need to render the electronic files and digital information unusable to an individual who improperly obtains a copy of the file or information.

SUMMARY OF INVENTION

The present invention is a system and method for securing and protecting a network environment and the information stored thereon from unauthorized use and sabotage. The invented system comprises: at least one remote workstation (PC) made up of any type of computer element typically engaged for computer network activity: i.e., desktops, laptops, other servers, etc., a centralized controlling server ("Netlatch server"), an always on communication link such as LAN, DSL, cable, or modem between the Netlatch server and the PC which can be transmitted through either public or private telecommunication lines, a power supply control system ("Powerlatch"), a network software application ("Netlatch"), a PC software application, scalable self executing applets ("Kets"), a Kets server which is communicably linked to a NetLatch server, and a secure file server which is also communicably linked to the NetLatch server. In the invention, the Netlatch server controls N number of PCs as described in Figures 2-4.

In operation, as a user attempts to access the network *via* a PC, the Powerlatch controls the power up process by controlling the power flow to the PC . If a proper authorization code is not received, then the power flow remains terminated and the PC remains disabled. If, however, a proper authorization code is received, a temporary supply of power is permitted, and the PC is allowed to do a partial boot up. Thereafter, resident software is loaded along with elements of an Operating System's (OS) networking software and another prompt is generated requiring yet another code or signature input directly from the user to verify the user and his security level.

After the user inputs the code or signature, the PC sends the code or signature to the NetLatch server for verification. The Netlatch server compares the code or signature with a security table which is predetermined by the owning organization of the network. If an improper code is entered, the PC shuts down. If a proper code is entered, the temporary power supply is converted to a continuous power supply and the PC examines the "Plug-n-Play"to verify proper function of the system components. If a proper function is verified the network allows the OS to completely launch. If an improper function is found, the PC shuts down.

Upon launching the OS, the latent controls are also launched along with predetermined applications corresponding to an authorized security level for the user. With respect to the latent controls, the Netlatch server receives periodic confirmations from the PC that the controls are operating correctly. If, at any time, the controls are found to be operating incorrectly, remedial action is taken by the Netlatch server.

As the user operates, all activity is performed through the Netlatch server. Since this required contact with the NetLatch server is always on, the Ket server is able to be in continuous contact as well. As file access is attempted, a routine initiates whereby the user's security level is compared to the files security level. If the levels match and the file is stored on the NetLatch server, the file is retrieved in a workable format. Alternatively, if the file is stored on the PC, the Netlatch server retrieves and sends an appropriate A-Ket to allow the file to be converted in a workable format. If the levels do not match, file access is denied or remedial action is commenced.

With respect to the storage of files, all files are saved either in the NetLatch server or locally are stored in an protected format using the Kets. Additionally, original back up files are stored in the secure server in a compressed format. The original back ups can not be accessed by any user. Rather, the protected format files can only be accessed and the files must be converted to a workable format before bing sent to the PC for use. As the file is re-saved, it is again converted into an protected format, this time using a different Ket.

The present invention is particularly useful because the NetLatch software has the capability of allowing access to a secure computer or file outside a secured building which houses the network so that workers can more easily access their work while away from the office. The NetLatch software also centralizes control over the network with what may have been previously independent applications. It centralizes control by bringing all network elements to one point, the NetLatch server.

The NetLatch software also allows enhanced firewall protection. Generally, network security arrangements have some type of firewall protection in place as part of the overall security system. NetLatch offers the ability to enhance the existing firewall application by requiring all communications to and from the remote PC to pass through the existing firewall. NetLatch is able to require this manner of communication because it has control of the PC *via* the constant monitoring of the latent controls. User's outside the private network could forgo this protection on non NetLatch servers because the communication routing described is not required in order to operate the PC. With NetLatch, the user must use the PC in accordance with the parameters set forth by the NetLatch server. Otherwise, the power flow will be terminated or other remedial action will be taken.

NetLatch also offers enhanced virus protection. This is both proactive and reactive in nature. NetLatch requires all users in the network to request permission to operate while on the PCs. It will be appreciated that there is a reactive example of enhanced virus protection from NetLatch. If a virus attacks a network using NetLatch, NetLatch allows an organization the ability, if they so choose, to shut down all its PCs until the virus is identified. Once identified, virus scanning software can be updated and then the PCs can be brought back onto the network after they have been scanned for the offending virus. This activity prevents any further damage from an unknown virus until it can be identified and stopped. NetLatch stops the spread and damage of the virus immediately, as soon as the virus is detected, thus minimizing loss of critical files while reducing the amount of time required to recover from the virus attack. Netlatch can proactively protect against viruses by requiring all PCs attempting to operate on the network to have an updated version of predetermined virus scanning software before allowing an unrestricted connection between the PC and the network.

NetLatch also provides asset-control capabilities. It can prevent PC theft because all the PCs are part of the NetLatch network and incapable of independent operation. Even if a PC is reported stolen, the NetLatch server can have it shut down permanently when access to the PC is attempted. Further, the NetLatch server, using Kets, can erase all the files or, leave them in a protected format, on the compromised machine, preventing unauthorized access to sensitive company information. It is also possible that the NetLatch server could learn the location of the missing PC, making it possible to recover the lost computing asset.

As an asset control device, NetLatch can give an organization control over the installed software base in its network of PCs. This can lead to productivity gains by the PCs' users. NetLatch can allow an organization the ability to standardize its software. For a given group functionality, NetLatch can ensure that only company-approved software is on those machines. This prevents lost time due to use of illegal software such as games, browsers, etc. An owning organization can input "best practice" parameters of approved software applications and NetLatch will enforce these parameters by prohibiting the use of other applications. Further, an individual or group can use NetLatch to minimize conflicts within the base of authorized software. They can troubleshoot problems, acquire patches or fixes, and automatically update all PCs in the network. This individual or group can also reduce lost time from major software upgrades. A test PC representative of the network can be upgraded. Any conflicts that occur can then be addressed before upgrading the rest of the network. This testing prevents lost time from individual PC users fixing duplicate problems on different machines. NetLatch offers software management that can be mandatory and therefore more effective. This leads to increased productivity gains by the PCs users.

NetLatch also functions as an asset management application. NetLatch allows an organization the ability to keep track of all computing hardware, software, and PC use. This can be valuable for accounting and insurance purposes, asset utilization studies, evaluation of employee work habits, and many other reasons.

Finally, NetLatch makes possible the implementation of Kets. Kets can effectively protect against the removal of files from a secured network without authorization. Moreover, Kets can prevent unauthorized access to files by individuals who have not successfully passed through the required validation steps or who do not have the required security level. This function is described in more detail below.

OBJECTS OF THE INVENTION

It is a principle object of the present invention is to provide a method and system for preventing theft and/or sabotage to electronic files or digital information by providing a secure network including scalable self executing applets attached to distributed files.

Another object of the present invention is to provide a method and system for restricting access to secure files on a network server.

Another object of the present invention is to provide a method and system for rendering stolen or copied files unreadable, and thereby useless, to an unauthorized user.

Another object of the present invention is to provide a method and system which controls a power supply to PCs through the use of a network server.

Another object of the present invention is to provide a method and system that continually monitors the status of all latent controls on a PC that is connected to a network thereby ensuring that all predetermined operating guidelines are followed.

Another object of the present invention is to provide a method and system for preventing theft of computer assets and to aid in the recovery of the same.

Another object of the present invention is to provide a method and system which requires numerous validation steps to verify that a user is authorized to enter a secure network and access secure files.

Another object of the present invention is to provide a method and system which requires all communications with a network server to be conducted through a fire wall.

Another object of the present invention is to provide a method and system for centralizing control of an entire network as opposed to employing independent applications with regards to security measures.

Another object of the present invention is to provide a method and system for enhanced virus protection.

Another object of the present invention is to provide a method and system which is capable of managing the use of an organization's assets such as PCs.

These and other objects of a claim eventual become apparent to ones skill in the art upon examination of the following drawings and detailed description.

BRIEF DESCRIPTION OF THE DRAWINGS

The forgoing and other objects will become more readily apparent by referring to the following detailed description and the appended drawing in which:

Figure 1 illustrates a diagram outlining the major components of the hardware usable in a typical PC system.

Figure 2 illustrates a diagram of a Star network environment according to the prior art.

Figure 3 illustrates a diagram of a Token Ring network environment according to the prior art.

Figure 4 illustrates a diagram of a Bus network environment according to the prior art.

Figure 5 illustrates a diagram outlining the major components of the hardware usable in the present invention.

Figure 6 is a flow chart of the boot up process according to the invention herein disclosed.

Figure 7 is a flow chart of the operation of a PC after boot up is complete

Figure 8 is a flow chart of the latent control operation check in process after boot up.

Figure 9 is a flow chart of the file retrieval process according to the present invention.

Figure 10 is a flow chart of the file storage process according to the present invention.

Figure 11 is a flow chart of the file storage process at the server according to the present invention.

Figure 12 is a pictorial diagram of the power supply component according to the present invention.

DETAILED DESCRIPTION

The invention herein disclosed is a network system 10 comprised of the following components: at least one remote workstation (PC) 12 made up of any type of PC element typically engaged for PC network activity: i.e., desktops, laptops, other servers, etc., a centralized controlling server ("Netlatch server ") 14, an always on communication link 16 such as LAN, DSL, cable, or modem between the Netlatch server 14 and the PC 12 which can be transmitted through either public or private telecommunication lines, a power supply control system ("Powerlatch") 18, a network software application ("Netlatch") 20, a PC software application 22, a scalable self executing applet application ("Kets") 24, a Kets server 26 which is communicably linked to the Netlatch server 14, and a secure file server 28 which is also communicably linked to the NetLatch Server 14.

The PC 12 provided for by the present invention is analogous to a typical PC system as identified in Figure 1. Figure 1 shows a system 200 that consists of a processor 202, for example a Pentium III™ processor by Intel for executing the instructions of currently running software. The processor 202 receives and places data on a bus 204, for example, Intel's Peripheral Component Interconnect, PCI bus which features wide (32/64 bit) data or instruction transfers at 120 MB per second. Active data or instructions are received and placed in a Random Access Memory, RAM 206. The memory 206 in a typical system is compound, that is it consists of a smaller very high speed cache portion an a larger, 64 M bytes typical, lower speed portion. Data and instructions placement are managed such that the performance of the combination approaches the speed of the cache with the size of the lower speed and lower cost technology. The data and instructions in the RAM 206 are volatile, that is said data and instructions are lost when the machine is powered down. A keyboard 208 is attached to the bus 204 for inputting keyed information. Vast amounts of non volatile data are routinely stored in today's systems on a Disk storage device 210. A disk storage device 210 consists of one or more platters that are coated with magnetic material and rotated at high speed, for example 7200 rpm, past a read/write head on an actuator. A disk storage 210 is really a subsystem for providing data to processor 202/ RAM 206. The technology has achieved the ability to economically provide 30 G-bytes and more data in the personally PC environment.

The attached point-and-click device 212 is typically a mouse. A trailing cursor (mark on the video screen) is manipulated in approximate correspondence with the position of the device 212.

The display 214 is really a visual subsystem with a complex set of electronics which include a refresh buffer that differs from conventional memory in the way it is ported. Video Ram is dual ported to allow simultaneous update of the image while continuously refreshing the display. Finally, a printer 218 and other peripheral devices 216 are connected to the system 200.

Network 10 access is provided in a number of different ways. In many systems, network access consists of a modem and telephone line connections. The Network 10 topology for this invention may employ various forms. These are illustrated in Figures 2-4. It is appreciated that the present invention may employ one or a combination of two or more of these environments. Figure 2 shows a "Star" environment 220 wherein a user may access the network 10 from a remote work station 12 by connecting through the use of the communication link 16. The communication link 16 first passes through a firewall 30 and then enters the centralized server 14. The centralized server 14 is connected to N units 12A-12C by a series of other communication links 16A-16C. The use of the communication link 16 through the firewall 30 is a mandatory flow of communication. Attempted accessed outside the firewall 30 cannot be achieved.

Figure 3 teaches a "token ring" network topology 222. The network 10 is substantially similar to the "Star" network 220 with the exception of the Token ring 224.

The standard for token ring protocol is expressed in Institute of Electrical and Electronics Engineers 802.5. Figure 4 teaches a "bus" network topology 226. In a bus network topology 226, the network 10 consists of a circuit arrangement wherein all devices are attached to a line, or bus 204 directly and all signals pass through each of the devices. Each device has a unique identity and can recognize those signals intended for it.

Figure 5 represents the PC according to the present invention. This arrangement is substantially similar to that of the typical computer system of Figure 1 except for an additional Powerlatch 18 device (described more completely below).

In the present invention, the Powerlatch 18 component is hardware based and controls the power flowing into the PC 12. As seen in Figure 12, powerlatch 18 is comprised of the following elements: a Powerkey 32, a Powerlock 34 and a communications port 38. The Powerkey 32 controls the powerlock 34 by signaling over a power line 36. The Powerkey 32 can control the Powerlock 34 by other methods as well. It shall be appreciated that the Powerkey 32 may also consist of a key card, a bar code, a password, numerical code, a biometric signature, a proximity sensor, or the like. The communication port 38 shall be fixed to the communication link 16 between the PC 12 and the NetLatch Server 14. The Powerlock 34 will be able to communicate to the PC 12 through the PC 12 to the NetLatch Server 14.

The NetLatch Server 14 operates *via* NetLatch 20. From the server 14, Netlatch 20 controls the entire network 10. One of Powerlatch's 18 functions is to extend Netlatch's 20 control over the network 10. This function is possible because of a continuous communication process between the NetLatch Server 14 and the PC 12. The NetLatch Server 14 has the ability to shut down any PC 12 that is operating through the network 10. NetLatch 20 also checks that an organization's network security "best practices" are followed and reports compliance to the NetLatch Server 14 by using Latent Controls 40 embedded in the PC 12. These Latent Controls 40 also report attempts to bypass the NetLatch Server 14 security.

Kets 24 are a series of security enhancements or additions in Netlatch 20 which are configured specifically to protect data from unauthorized access or dissemination. Kets 24 function is to make all data access dependent upon the network 10 and guidelines set forth by an organization. More specifically, Kets 24 are scalable self executing applets which attach to sensitive files 1 within the NetLatch Server 14. Kets 24 are stored and provided by a Kets Server 26 which is always communicably linked to the NetLatch Server 14. A Kets 24 program can be configured individually to perform a variety of tasks in defense of the network 10. A Kets 24 response to an intrusion can be anything from a simple warning up to erasing secure files moved to an intruding PC, or worse.

Kets 24 are a proactive and reactive defense for sensitive data housed in the network 10. Kets 24 are proactive because they act as a strong deterrent to electronic data theft. Kets 24 are just-in-time reactive because they respond to electronic intrusion automatically and immediately.

The use of Kets 24 are possible because of the NetLatch 20 and NetLatch Server 14. Because all the PCs 12 in the network 10 are tied to the NetLatch Server 14, the Kets Server 26 can be employed to work through the NetLatch Server 14 and perform a Kets Security function. Through the NetLatch Server 14, the Kets Server 26 automatically attaches appropriate Kets 24 to all Kets-level security files in the Network 10. Kets 24 can be attached to files in various methods. In one embodiment, all files in the Network 10 are converted from their application's format into a proprietary Kets file format (KFF). The KFF has integrated into the stored data a Kets 24.

As shown more completely in Figure 10, at the point when a file is going to be stored 42, the NetLatch Server 14 requests a new Kets 24, 44 from the Kets Server 26. Once the Kets 24 is retrieved, it is attached to the file 46. Thereafter, an encryption constant is requested 48 and the file is encrypted / converted to a KFF 50. If the file is to be stored locally 52, NetLatch 20 inquires as to whether the file is to be compressed 54. After receiving an appropriate response, the file is either compressed 56 and stored 58, or stored as is 58 according to Figure 11. If the file is to be stored on the server 14, it is automatically compressed and forwarded 60 from the PC 12 to the NetLatch Server 14. Figure 11 shows the method of storing a KFF file in the NetLatch Server 14. After receiving the file, NetLatch 20 decompresses it 64. Once the file is decompressed, it must be converted / decrypted from the KFF 66. To do this the corresponding A-Kets 68 is retrieved and run 70. After the file is converted back to the normal file format, it is saved 72A and 72B respectively. Once converted, the original file is deleted. In order to access any secured file in the Kets format, the file has to be converted back into its original format from the KFF. To convert the file back from the KFF the Kets Server 26 is required to provide the correct A-Kets 68 to remove the Kets 24 integrated into the file. If the KFF is opened without the proper A-Kets 68, the KFF file destroys itself and the application's file before it can be accessed. If the A-Kets 68 is present, the Kets 24 is erased and the file is converted back into the original application's format for the user's authorized access.

In another embodiment, instead of converting the files to KFF (or in conjunction with), the Kets 24 could convert a directory drive of the PC 12 drive containing the secure files. The Kets Server 26 would create a false directory of files on the converted drive. As soon as the PC 12 is booted up or a Kets-secured disk is inserted in the disk drive, this false directory is loaded and executed. This method activates the Kets 24 before any attempt to open a specific file. When the false directory is executed the Kets 24 then erase the contents of any or all parts of the drive. If the PC 12 is properly online with the NetLatch 14 and Kets Servers 26 the false directory is circumvented.

In still another embodiment, the Kets Server 26 can directly convert the files in transfer from the Secure File Server 28 and/or in transit from one machine to the next or in communications between the NetLatch Server 14 and the PCs 12 in the network 10.

In addition to the various methods of use for Kets, Kets are capable of numerous functions. In one embodiment, a Reactive Kets is used. This type of Kets is similar to traditional viruses, except the Kets does not replicate itself and /or spread. This type of Kets when triggered can issue a warning to the user, erase the file, erase the entire disk(s), or even corrupt the PC 12 system in use to prevent the file's disclosure to the unauthorized access attempt. The severity of the Kets reaction is dependent upon the organization employing the Kets for file security.

Kets can act in forms other than simply erasing files. Kets could be part of compression and encryption routines. When a user accesses a compressed or encrypted secure file, the Kets could simply change the encryption key or compression algorithm in a manner that when the decryption or decompression routine is finished, the resulting data is jumbled or garbage.

Another embodiment would be Access Kets. These Kets simply prevent access to the file. They may be triggered by unauthorized PC 12 activities such as copying, electronic transfer, data manipulation, etc. Access Kets can allow files of different security clearances on the same PC, 12 at the same time. The Access Kets would allow or deny access to a file based on the already known security level of the user. The Kets system may employ encryption to this effect. Only if the user is authorized access by the NetLatch Server 14 will the NetLatch Server 14 provide the necessary decryption Key to open the encryption file, even if the file is stored on the remote user's PC 12.

Another embodiment is the Whistleblower Kets. The Whistleblower Kets does what its name implies: reports any unauthorized file access or activities to a proper authority. The Whistleblower Kets operates as a silent alarm, thereby not alerting the user that activity has been detected and is being reported. The Whistleblower Kets can report illegal activity in a number of ways. First, it can report directly to the NetLatch Server 14 if available. If the illegal activity is happening off the Network 10, the Whistleblower Kets can report indirectly *via* other communication channels that may be available. Finally, the Whistleblower Kets can report through standard contagious viral activities. The Whistleblower Kets can encode a file with a harmless viral strain containing the user, file, date, time, and nature of the intrusion. Also encoded into this virus would be a signature that only a Kets Security PC 12 would know to scan for. The Whistleblower Kets then spreads in a harmless fashion. It spreads from PC to PC until eventually the Kets Security PC 12 that monitors for such benign Whistleblower Kets signatures would find it and decode the information.

Another embodiment is the Communication Kets 24. Communication Kets 24 are designed to secure communications between two points over either a secure or unsecured communications link. Communication Kets 24 can increase protection beyond the level provided by encryption. Before communicating a file, the Kets Server 26 converts the file to KFF. The file is then transmitted and on the receiving end another Kets Server 26 reconverts the transmitted file with an appropriate A-Kets 68. The Kets Servers 26, operating independently of each other, are capable of identifying what Kets 24 is part of the transmission. By converting the communicated file in this manner, any unauthorized PC 12 that is monitoring the communications link 16, will only be able to, download the KFF. Once downloaded the Kets 24 can act in any number of ways, as outlined earlier, to protect that file from being accessed on the unauthorized PC 12.

Another aspect of Kets 24 is that they can be nested--that is, more than one Kets 24 can protect a file. In order to illegally access the data file each Kets 24 would have to be bypassed. Any one Kets 24 that is triggered prevents successful access. Nesting Kets 24 makes protecting highly sensitive files very effective. When nested, some Kets 24 may even be silent when triggered.

The use of Kets 24 are effective for security purposes for a variety of reasons. Indeed, Kets 24 are continually updated, stockpiled, and distributed randomly by the Kets Server 26 through the Network 10. Kets 24 that have been overused or compromised are removed and replaced with new, virgin Kets 24. Kets 24 are collected for use on the Kets Server 26. The higher the number of Kets 24 that are stockpiled, the higher the effectiveness of the Kets 24 system. The Kets Server 26 then automatically distributes the stockpiled Kets 24 in a random manner. Only the Kets Server 26 knows which Kets 24 is attached to which file. Each time the PCs 12 contacts the NetLatch Server 14, the old Kets 24 are removed and replaced with new Kets 24.

In operation and as illustrated in Figures 6-9, a user attempts to power up the PC 12 at step 74. At this time the user must use the proper Powerkey 32 by inputting it to the Powerlock 34, step 76. The Powerkey 32 is a non-repeating code as previously defined. An example of a non-repeating key is described in U.S. Patent 5,530,431. Next, the Powerlock 34 allows a temporary power flow to the PC 12, step 78. The Powerlock 34 is installed into the PC 12 either internally or externally. The Powerlock 34 operates as a switch which initially prevents power from flowing to the PC 12 from power lines 36 which are connected to a power supply. Powerlock 34 also maintains the capability of terminating the power to a PC 12 if a user attempts to operate the PC 12 without NetLatch Server 14 access authorization. The Powerkey 32 is kept in the authorized user's possession. It is appreciated that the Powerkey 32 can have many embodiments as previously described. Immediately after the power up 78, the POST and BIOS setup begin 80. However, before the POST and BIOS setup can complete, the remote NetLatch 20 software seizes control of the PC 12.

At this time, the user is prompted for a password or other form of validation 82. The user must then input the required information. This required input can consist of a password, a PC Id, a key card, or a biometric signature. Subsequent to inputting the information, the PC 12sends it (the input data and the Powerkey 32 data) to the NetLatch Server 14 for review 84. After receiving the information, the NetLatch Server 14 conducts a comparison 86 to corresponding and pre-stored data which includes the users security level. If there is a match, the PC 12 proceeds to conduct a check of the "Plug-n-Play" 88. If there is no match, the PC 12 shuts down 90.

The term "Plug-n-play", as it is used in this application, refers to a capability for the OS that gives the user the ability to plug a device (i.e., keyboard, mouse, Internet devices, etc.) into the PC 12 and have the PC 12 recognize that there is a device there. The user doesn't have to independently tell the PC 12 that the device is present.

After obtaining the status of the "Plug-n-Play", the PC 12 sends the information to the NetLatch Server 14 along with other circumstances including but not limited to the users identity, the user's password, and the workstation's software status. Upon receipt, the NetLatch Server 14 compares the circumstances to a predetermined and stored set of guidelines (as provided by the network's owning organization). If the circumstances are in compliance with the predetermined guidelines, the OS is launched 92.

Subsequent to the OS being launched 92, the latent controls 40 are also launched 94. These controls 40 monitor the launching of all software applications 96, the launching of clipboard use 98, a periodic check-in 100 requirement of the PC 12 to the server 14, and a data modification 102 or transfer monitor 104. Additionally, any locally stored navigator or browser which is permitted by the network 10 is also launched 106. As the user navigates through the available applications, several file requests may be made 108. The file requests 108 can be made either to the server 14 or to the local PC 12, 112 and 114 respectively. Regardless of where the file request 108 is made, the same process is followed as the PC 12 is linked and under the control of the NetLatch Server 14.

Once a file request 108 is made for files stored on the server 14, the NetLatch Server 14 reviews the circumstances presented from the PC 12 and evaluates whether or not the user has an acceptable security level 116 which will allow use of the file. If the user does not have the necessary security level, access is denied 118. If, however, the user has the required security level and authorization, the NetLatch Server 14 then checks to see if the file is in use 120 and 122. If the file is stored on the NetLatch Server 14 and being used by another user, a message is sent 124 to the user inquiring if a read only format file is acceptable for use 126. If a read only format is acceptable, the NetLatch Server 14 retrieves the file 128, retrieves the corresponding A-Kets 130, decrypts the file 132 and sends it to the user 134 as a read only file. If, however, the file is stored on the PC 12, then the server 14 checks the security level 116, and if the appropriate level of authorization exists, the server 14 retrieves the A-Kets 130, decrypts the file 132 and sends it to the user 134. If the file is then saved on the PC 12 the NetLatch Server 14 provides a Kets 24 to protect that file if reopened by a user in the future.

With respect to the latent controls 40, NetLatch 20 continually requires confirmations that said controls 40 are in compliance with all the guidelines that have been set forth for the network, 136, 138, 140, 142, 144 respectively. As illustrated in more detail in Figure 8, if at any time, an initial application is opened and thereafter falls out of compliance with the pre-determined guidelines, remedial action 146 is taken. The remedial action taken 146 can be any one of the actions defined above. If the controls 40 are in compliance, the permitted action occurs 148, 150, 152, 154, 156 respectively.

In regard to Figure 8, The PC 12 contacts the NetLatch Server 14 every xx seconds with a code that is generated by an algorithm that is unique to the PC, the time, the network card, hardware number, the last encryption key string and the specific encryption program sent to that PC 12 xx seconds ago by the NetLatch Server 14. The NetLatch Server 14 monitors this and, if a PC 12 fails to check in on time with precisely the right answer, creates a network manager warning. This "action on absence" process combined with transient security keys quickly senses when someone has booted from a floppy or other device. Additionally, it makes it impossible for a user to go to another PC on the network 10 pretending to be the PC that has been taken out of service for hacking. Additionally, this process, along with a "sniffer" function on the NetLatch Server 14 to identify new IP addresses talking, also makes it impossible to plug in a cheap hub and laptop to bypass copy controls.

Additionally, the PC 12 notifies the NetLatch Server 14 of all copies, email send file attempts and other suspicious behavior sensed with means described below. The NetLatch Server 14 keeps a journal file and responds to statistics (such as heavy file access or sensitive area access). NetLatch 20 prevents and intercepts (using low level local sensing and real time NetLatch Server 14 directed permission/denial) the copying to any unauthorized device (which may be conditional based on user, device or type of file being copied) or attachment to email of sensitive files to email and notifies the NetLatch Server 14 of the attempt. Through the monitoring of the latent controls 40, NetLatch 20 can also prevent unauthorized loading of new drivers/services of any kind. NetLatch 20 is capable of sensing and preventing the copy and reports to the user a device error. NetLatch 20 can also prevent execution of any program either not on an approved list or on a "don't copy" security list by monitoring the latent controls 40. This approved list/table could reside and be updated only on the NetLatch Server 14. This same table could contain acceptable or unacceptable devices for each file mentioned. NetLatch 20 will also verify (by reading key bytes and checksums from the file to be executed and comparing it to values stored in the NetLatch Server's 14 table) that the named program is actually the approved program.

SUMMARY OF THE ACHIEVEMENTS OF THE OBJECTS OF THE INVENTION

From the forgoing it is readily apparent that I have invented a method and system which prevents theft and/or sabotage of an electronic file or digital information stored in a network by providing defensive scalable self executing applets attached to the files or information. It is further apparent that I have invented a method and system for restricting access to secure files on a Network, rendering stolen or copy files unreadable, and thereby useless, to an unauthorized user, and controlling a power supply to our PCs through the use of a Network Server. It is further apparent that I have invented a method and system whereby numerous validation steps to verify that a user is authorized to enter a secure network are required. Still further, I have invented a method and system which requires all communications with the Network Server to be conducted through a fire wall while allowing the centralized control over the Network.

It is to be understood that the foregoing description and specific embodiments are merely illustrative of the best mode of the invention and the principles thereof, and that various modifications and additions may be made to the system and method by those skilled in the art, without departing from the spirit and scope of this invention.

What is claimed:

1. A method of protecting a computing network environment and the electronic data stored thereon from unauthorized use by a user comprising the steps of:

providing a uniquely formatted electronic data file on a source computer through a remote workstation in a network;

providing a power supply component of the remote workstation with a reader, said reader permitting powering-up the remote workstation only upon receipt of a unique code;

providing a transmitting device for transmitting the unique code to the reader, wherein the reader permits powering-up of the remote workstation only if the unique code is accepted;

at the remote workstation, establishing communication between the remote workstation and the source computer;

at the source computer, initializing a first set of instructions and sending said first set of instructions to the remote workstation;

at the remote workstation, receiving the first set of instructions from the source computer, requiring the user to verify a first prescribed set of circumstances and sending the first set of circumstances to the source computer;

at the source computer, receiving the first set of circumstances, verifying them, and sending a signal to the remote workstation to launch an operating system;

at the remote workstation, launching the operating system and launching a series of software applications for navigation by a user;

at the remote workstation requesting access to the uniquely formatted electronic data file and sending the request the source computer;

at the source computer, receiving the request and initializing a second set of instructions and sending the second set of instructions to the remote workstation;

at the remote workstation, receiving the second set of instructions from the source computer, responding to the second set of instructions by providing a second set of circumstances and forwarding the second set of circumstances to the source computer;

at the source computer, receiving the second set of circumstances, verifying them, and fetching the requested file if the second set of circumstances are properly verified; and

at the source computer, launching access to the uniquely formatted electronic data file by reformatting the file into a readable text format and sending the file from the source computer to the remote workstation for use.

2. The method according to claim 1 wherein the uniquely formatted electronic data file is a converted text file with a scalable self executing applet embedded thereto which, if accessed without being reconverted to a readable text file, deletes itself.

3. The method according to claim 1 wherein the uniquely formatted electronic data file is a converted text file with a scalable self executing applet embedded thereto which, if accessed without being reconverted to a readable text file, corrupts the remote workstation's hard drive.

4. The method according to claim 1 wherein the uniquely formatted electronic data file is a converted text file with a scalable self executing applet embedded thereto which, if accessed without being reconverted to a readable text file, erases the entire disk(s) in which the electronic data file is stored.

5. The method according to claim 1 wherein the uniquely formatted electronic data file is a converted text file with a scalable self executing applet embedded thereto so that if the electronic data file is attempted to be accessed without first being reconverted to a readable text format access is denied.

6. The method according to claim 1 wherein the uniquely formatted electronic data file is a converted text file with a scalable self executing applet embedded thereto so that if the electronic data file is accessed without first being reconverted to a readable text format a whistleblower algorithm is activated which reports the improper access to a predetermined authority.

7. The method according to claim 1 wherein the power supply component is provided internally to the remote workstation.

8. The method according to claim 1 wherein the power supply component is provided externally to the remote workstation.

9. The method according to claim1 wherein the unique code is a password.

10. The method according to claim 1wherein the unique code is a bar code.

11. The method according to claim 1 wherein the unique code is a biometric signature.

12. The method according to claim 1 wherein the first set of circumstances is a password.

13. The method according to claim 1 wherein the second set of circumstances consists of the unique code, the first set of circumstances, and a status indicator of the remote workstations hardware and software.

14. The method according to claim 1 wherein the remote workstation is a desk top computer.

15. The method according to claim 1 wherein the remote workstation is a lap top computer.

16. The method according to claim 1 wherein the remote workstation is a computer network server.

17. A system for providing security to a computing network environment and electronic data stored thereon comprising:

a centralized server capable of working in a network environment having at least a memory device and a series of communication ports for receiving communication connections;

means for operating said centralized server programmed into a CPU of said centralized server;

at least one electronic data file capable of being stored on said centralized server in said memory device;

at least one remote workstation having an associated power supply;

said remote workstation being capable of working in a network environment and having a power supply control means for controlling a power flow to said remote workstation and a communication means for communicating with said centralized server, said remote workstation being communicably linked to said centralized server;

means for operating said remote workstation programmed into a CPU of said remote workstation;

means for embedding a plurality of scalable self executing applets to said electronic files, said applets being contained in an applet server which is communicably linked to said centralized server;

means for removing said scalable self executing applets from said electronic files, said means for removing said applets being stored in said applet server;

a secure file server for storing electronic files which are not embedded with said applets, said secure file server being communicably connected to said centralized server; and

wherein said centralized server controls all remote workstations which are communicably linked thereto by said means for operating said centralized server.

18. The system according to claim17 wherein said applet is a low-level code which prohibits access to said electronic file if a set of circumstances is not produced.

19. The system according to claim17 wherein said applet is a low-level code which deletes said electronic file if access is attempted by an unauthorized user.

20. The system according to claim17 wherein said applet is a low-level code which

corrupts the means for operating the remote workstation if an unauthorized use is attempted.

21. The system according to claim17 wherein said applet is a low-level code which reports an improper use to a predetermined authority.

22. The system according to claim17 wherein said network environment resembles a Star topology.

23. The system according to claim17 wherein said network environment resembles a Token topology.

24. The system according to claim17 wherein said network environment resembles a bus topology.

25. The system according to claim17 wherein said communication means is a public telecommunication line.

26. The system according to claim17 wherein said communication means is a private telecommunication line.

27. The system according to claim17 wherein said power supply control means comprises a reader and a corresponding unique input code.

28. The system according to claim 27 wherein said unique input code is a password.

29. The system according to claim 27 wherein said unique input code is a bar code.

30. The system according to claim 27 wherein said unique input code is a biometric signature.

31. A method for providing security to a computer network and a plurality of files stored thereon, comprising the steps of:

a. providing at least one network server for hosting at least one remote workstation computer;

b. providing a power supply component capable of restricting a power flow to said computer consisting of an input device containing an input code and a corresponding reader, which when unlocked provides the power flow to the computer;

c. setting a predetermined code for unlocking the power flow;

d. providing a means for verifying that the input code and the predetermined code match;

if said input code and said predetermined code match, then performing the following steps:

e. launching an operating system on said computer;

f. prompting for a user input;

g. providing means for verifying that the user input is authorized;

h. launching latent controls;

i. gathering status of latent controls;

j. sending status to the network server;

k. providing a means for verifying status of latent controls is in compliance with a predetermined set of circumstances;

iteratively repeating steps i-k until the status of the latent controls are out of compliance with said set of predetermined circumstances; and terminating power flow to computer; and

l. terminating the power flow to the remote workstation computer.

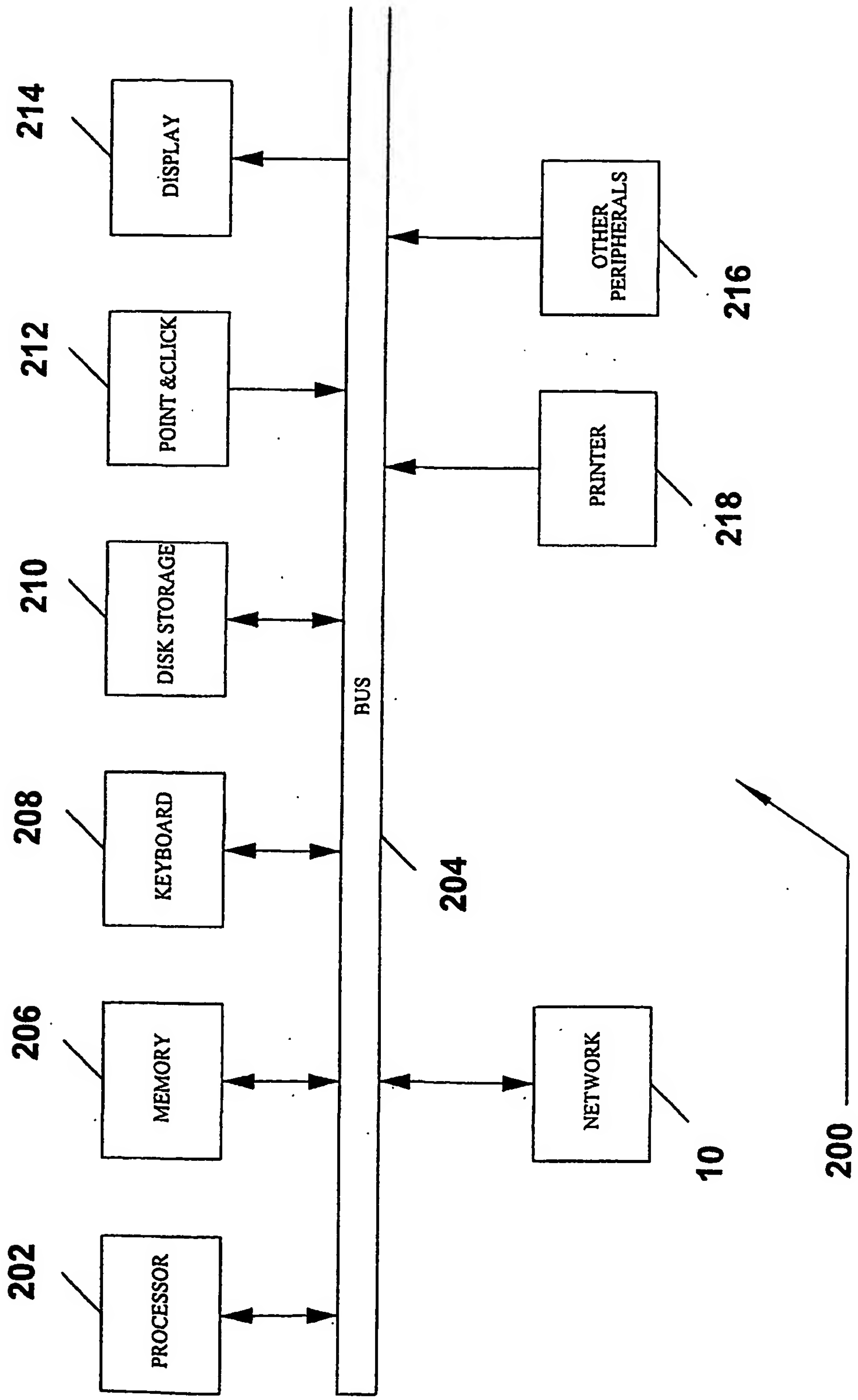

PROCESSOR
202
MEMORY
206
KEYBOARD
208
DISK STORAGE
210
POINT &CLICK
212
DISPLAY
214
BUS
204
NETWORK
10
PRINTER
218
OTHER PERIPHERALS
216
200

Figure 1

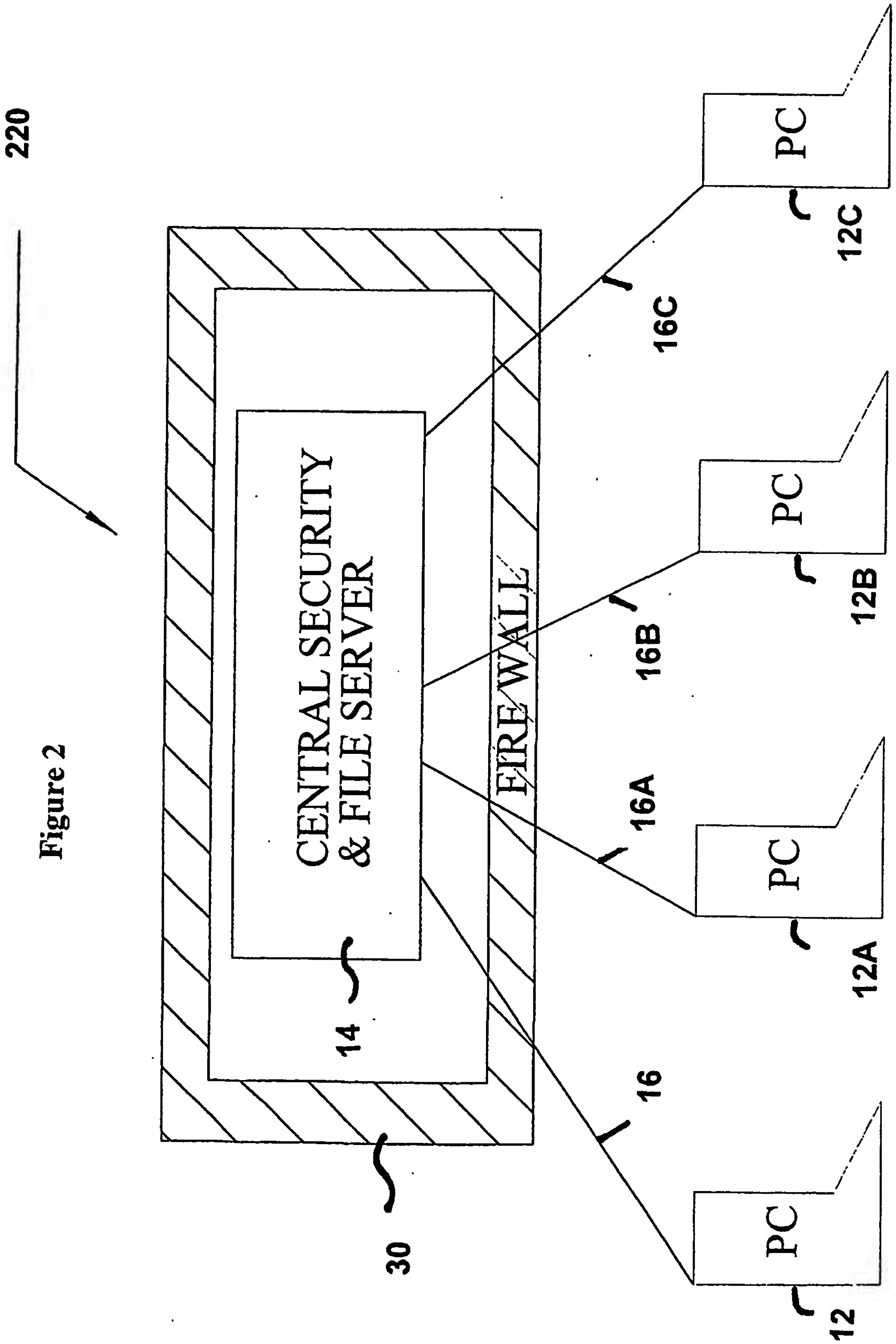

220
CENTRAL SECURITY
& FILE SERVER
FIRE WALL
14
30
16
16A
16B
16C
PC
PC
PC
PC
12
12A
12B
12C

Figure 2

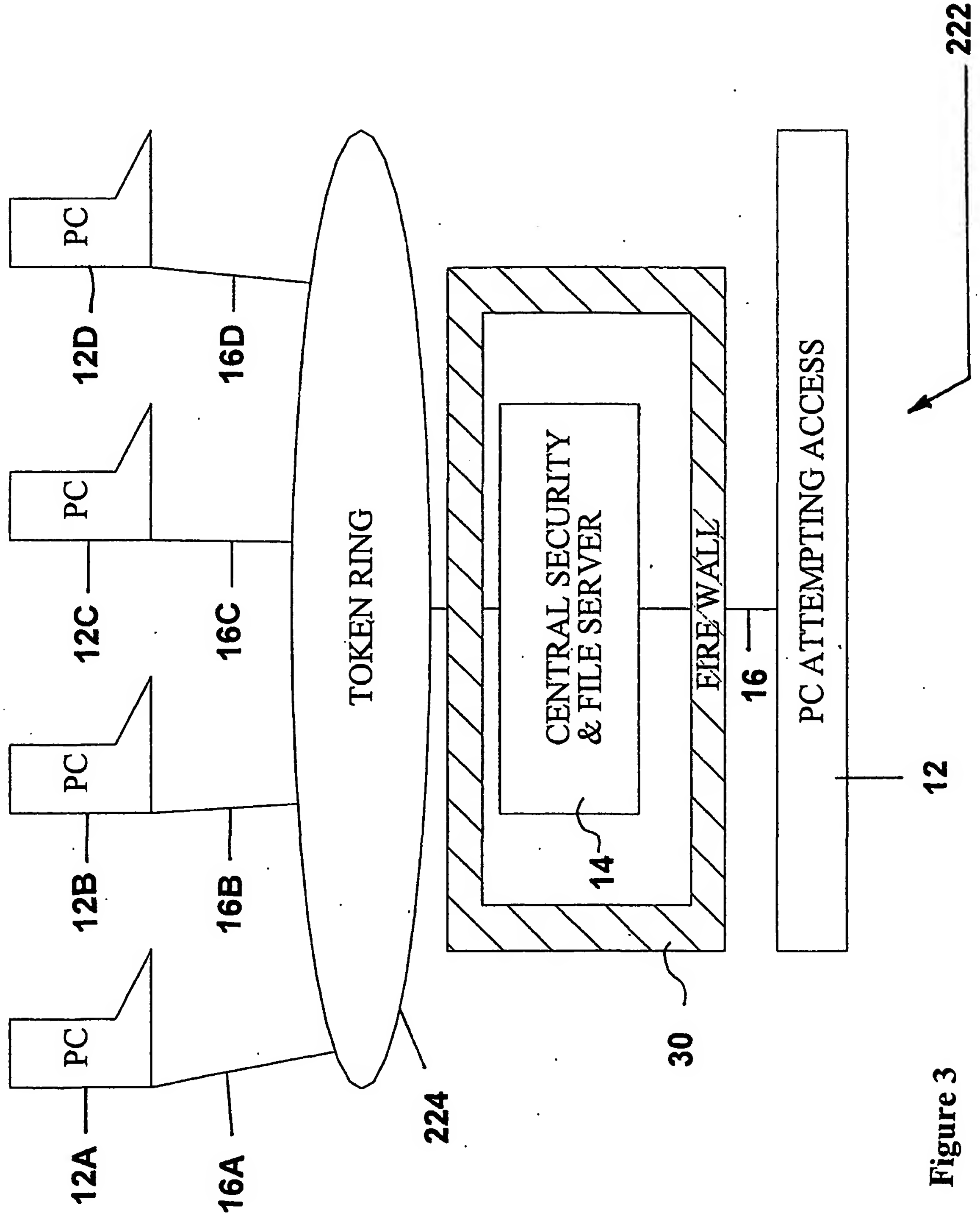

PC
PC
PC
PC
12A
12B
12C
12D
16A
16B
16C
16D
TOKEN RING
224
CENTRAL SECURITY & FILE SERVER
14
FIRE WALL
30
16
PC ATTEMPTING ACCESS
12
222

Figure 3

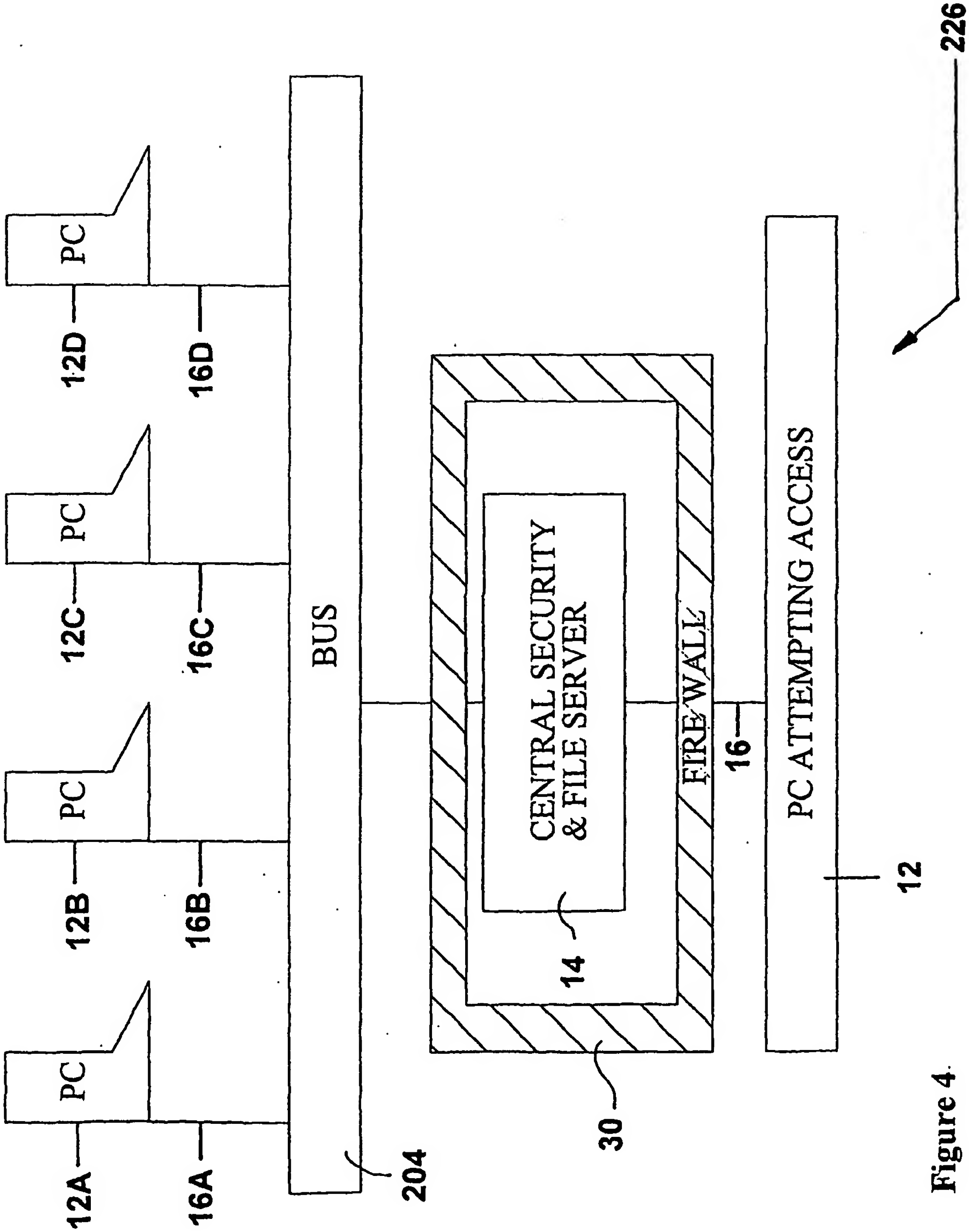

PC
PC
PC
PC
12A
12B
12C
12D
16A
16B
16C
16D
BUS
204
CENTRAL SECURITY & FILE SERVER
14
FIRE WALL
30
16
PC ATTEMPTING ACCESS
12
226

Figure 4

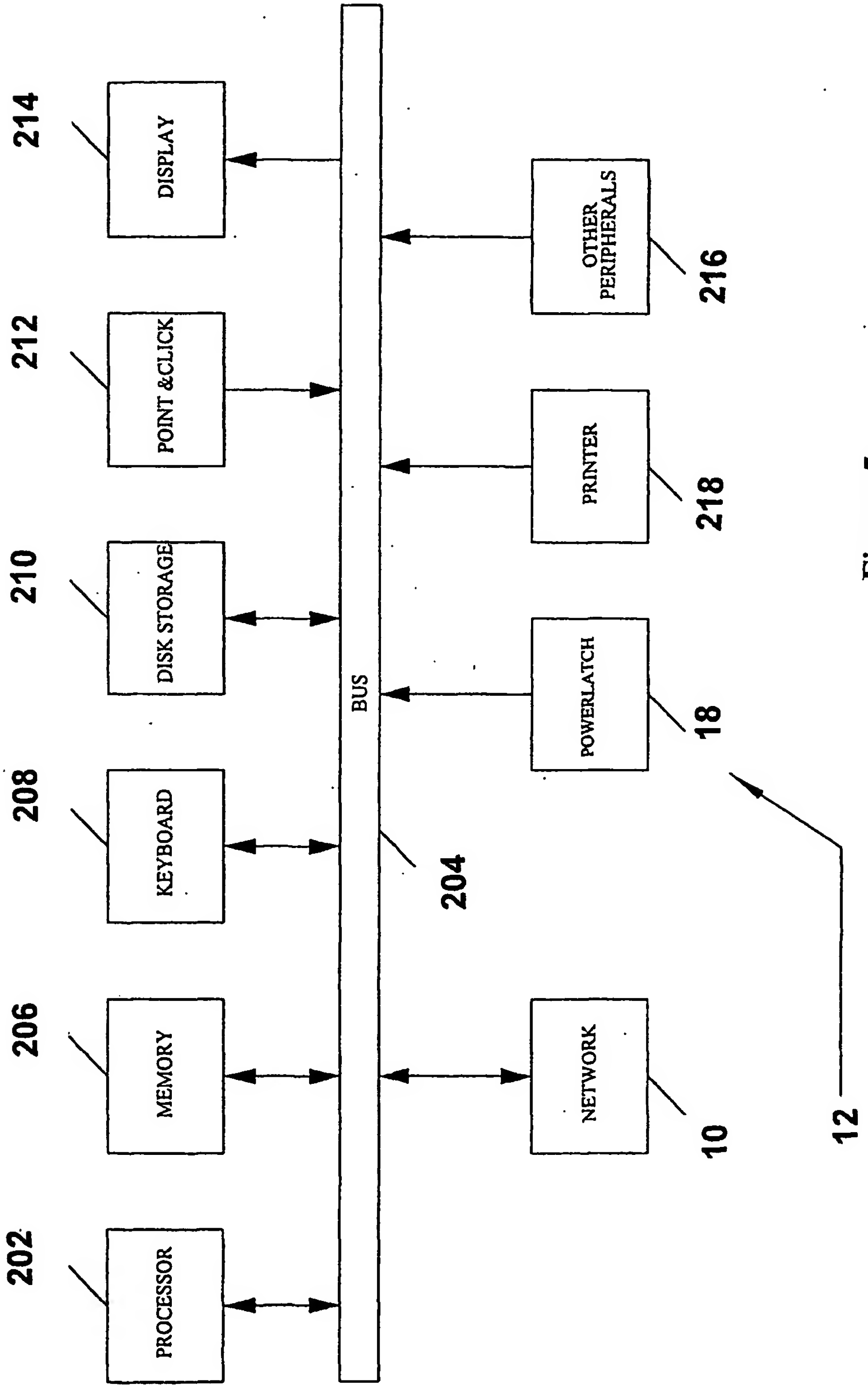

214
DISPLAY
212
POINT &CLICK
210
DISK STORAGE
208
KEYBOARD
206
MEMORY
202
PROCESSOR
BUS
204
OTHER PERIPHERALS
216
PRINTER
218
POWERLATCH
18
NETWORK
10
12

Figure 5

Figure 6

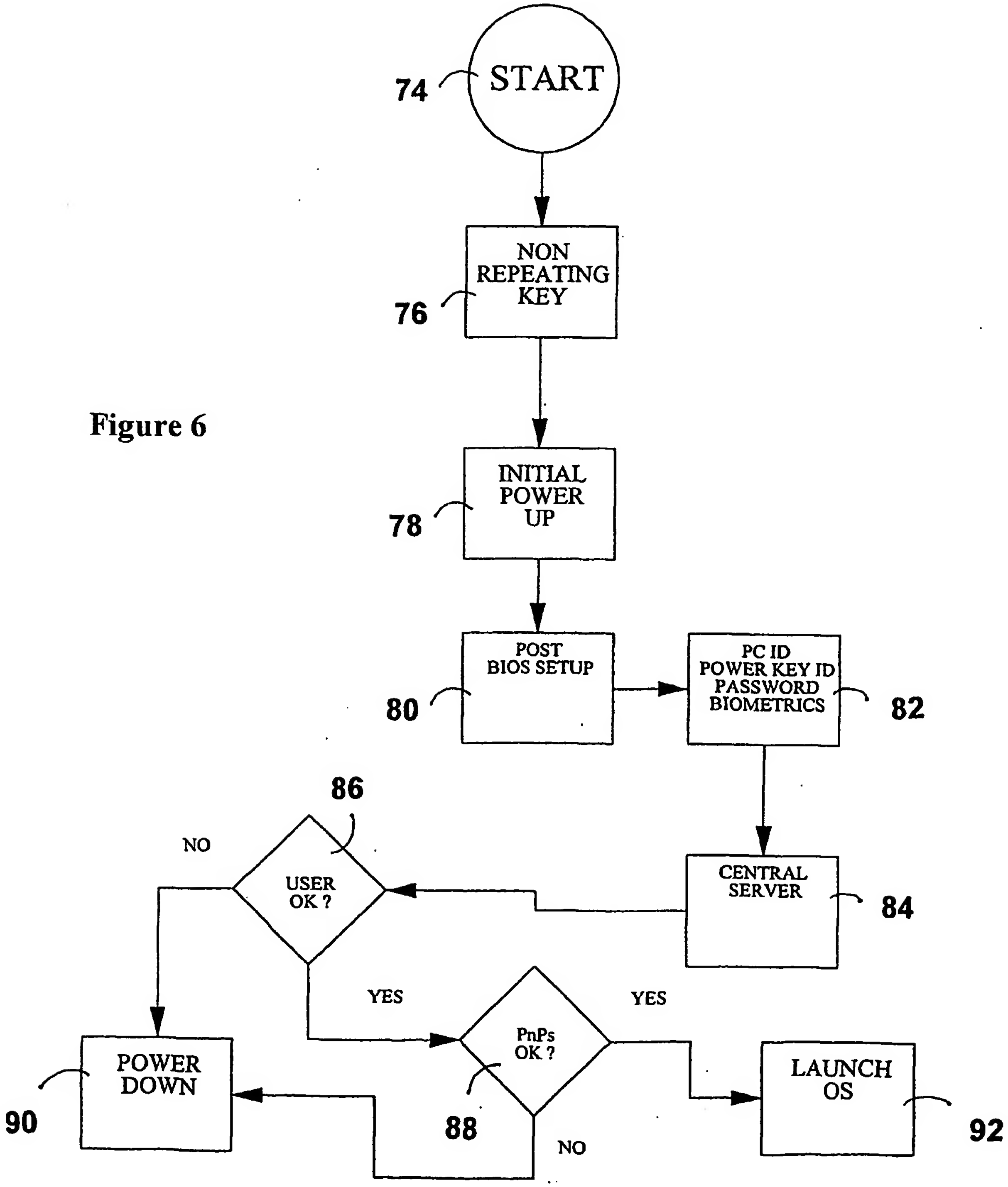

START
74
NON REPEATING KEY
76
INITIAL POWER UP
78
POST BIOS SETUP
80
PC ID POWER KEY ID PASSWORD BIOMETRICS
82
CENTRAL SERVER
84
USER OK ?
86
NO
YES
PnPs OK ?
88
YES
NO
POWER DOWN
90
LAUNCH OS
92

Figure 7

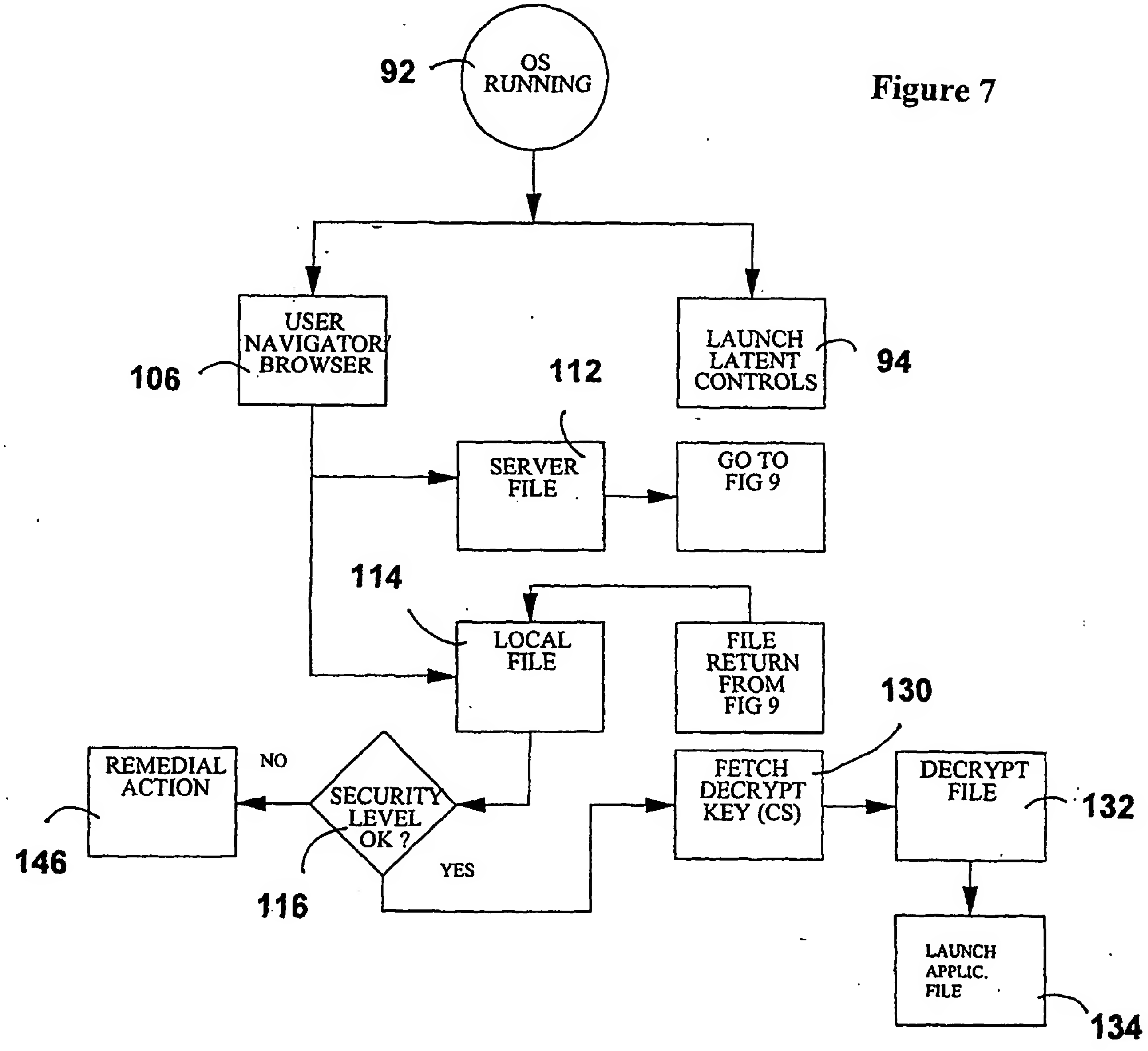
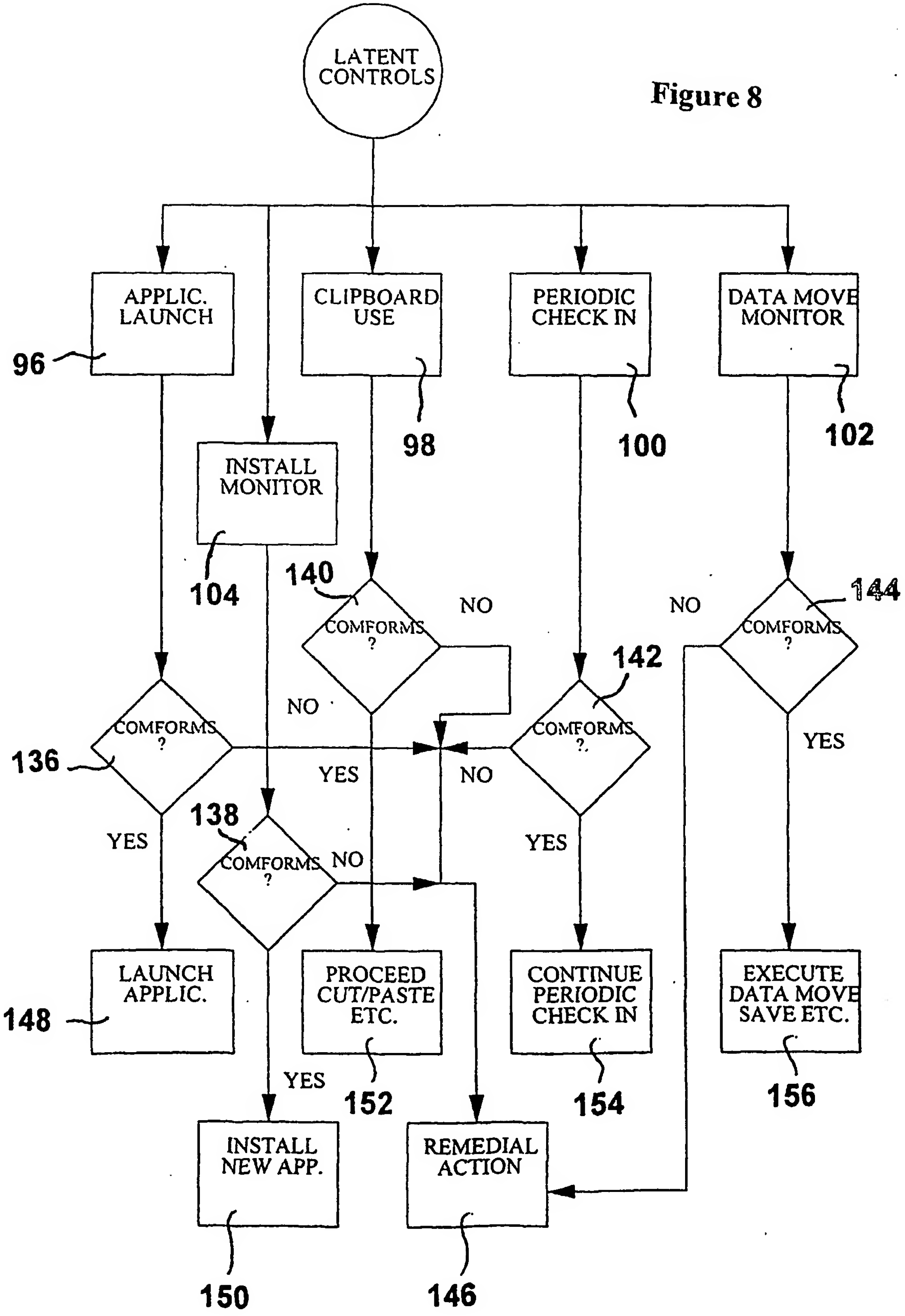

92
OS RUNNING
USER NAVIGATOR/ BROWSER
106
LAUNCH LATENT CONTROLS
94
112
SERVER FILE
GO TO FIG 9
114
LOCAL FILE
FILE RETURN FROM FIG 9
130
REMEDIAL ACTION
NO
SECURITY LEVEL OK ?
YES
FETCH DECRYPT KEY (CS)
DECRYPT FILE
132
146
116
LAUNCH APPLIC. FILE
134

Figure 8


LATENT CONTROLS
APPLIC. LAUNCH
96
CLIPBOARD USE
98
PERIODIC CHECK IN
100
DATA MOVE MONITOR
102
INSTALL MONITOR
104
140
COMFORMS ?
NO
NO
144
COMFORMS ?
142
COMFORMS ?
NO
136
COMFORMS ?
YES
NO
YES
YES
YES
138
COMFORMS ?
NO
YES
LAUNCH APPLIC.
148
PROCEED CUT/PASTE ETC.
152
CONTINUE PERIODIC CHECK IN
154
EXECUTE DATA MOVE SAVE ETC.
156
YES
INSTALL NEW APP.
150
REMEDIAL ACTION
146

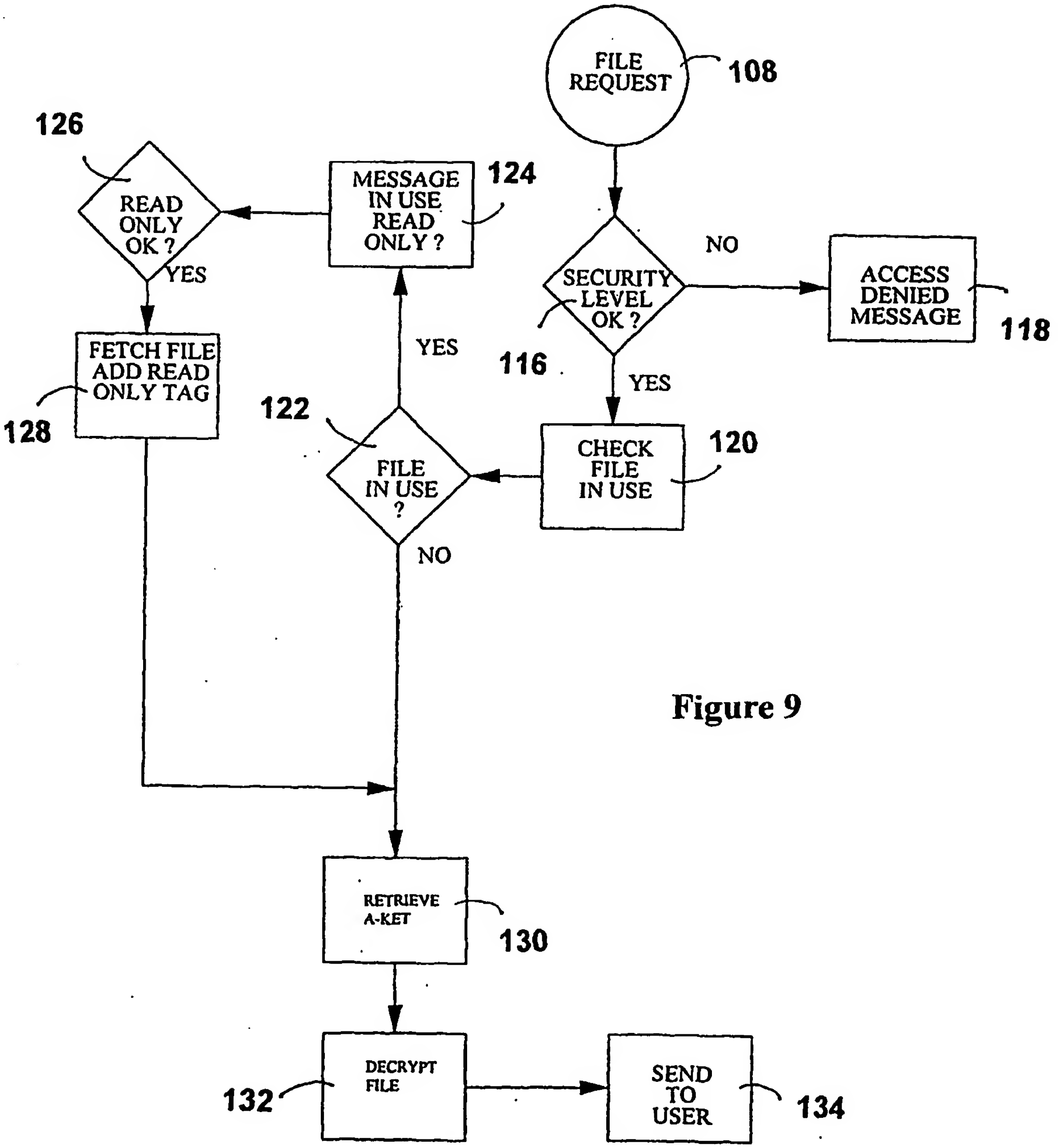

FILE REQUEST
108
SECURITY LEVEL OK ?
116
NO
ACCESS DENIED MESSAGE
118
YES
CHECK FILE IN USE
120
FILE IN USE ?
122
YES
NO
MESSAGE IN USE READ ONLY ?
124
READ ONLY OK ?
126
YES
FETCH FILE ADD READ ONLY TAG
128
RETRIEVE A-KET
130
DECRYPT FILE
132
SEND TO USER
134

Figure 9

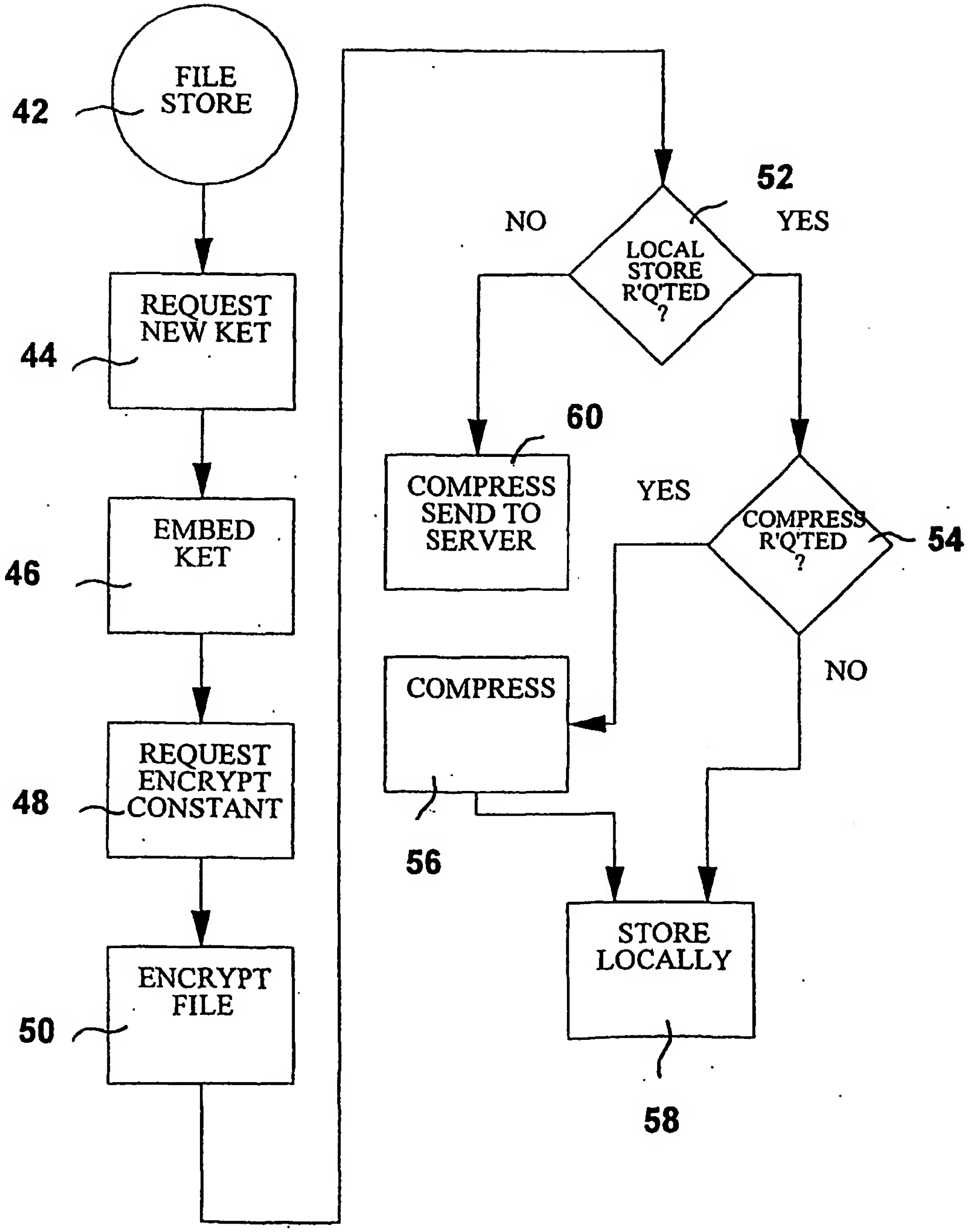

FILE STORE
42
REQUEST NEW KET
44
EMBED KET
46
REQUEST ENCRYPT CONSTANT
48
ENCRYPT FILE
50
52
NO
YES
LOCAL STORE R'Q'TED ?
60
COMPRESS SEND TO SERVER
YES
COMPRESS R'Q'TED ?
54
NO
COMPRESS
56
STORE LOCALLY
58

Figure 10

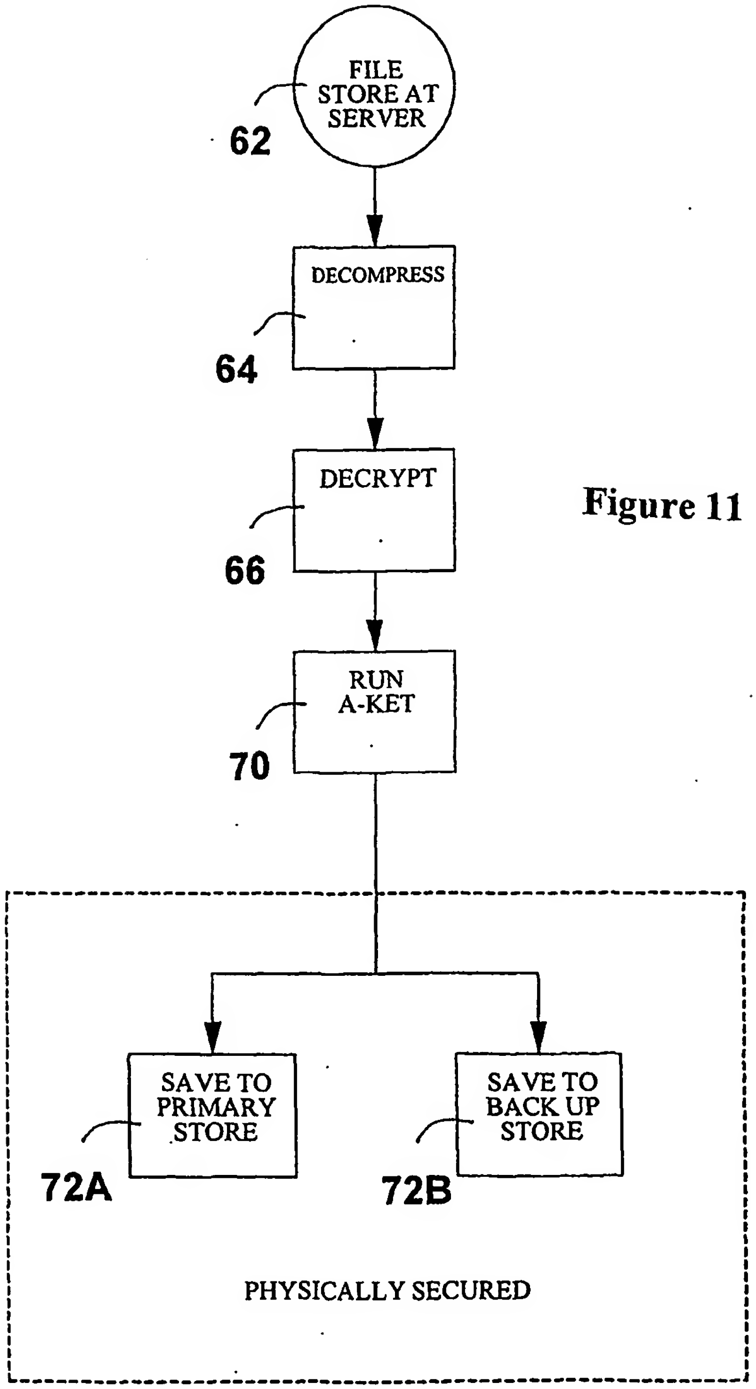

FILE STORE AT SERVER
62
DECOMPRESS
64
DECRYPT
66
RUN A-KET
70
SAVE TO PRIMARY STORE
72A
SAVE TO BACK UP STORE
72B
PHYSICALLY SECURED

Figure 11

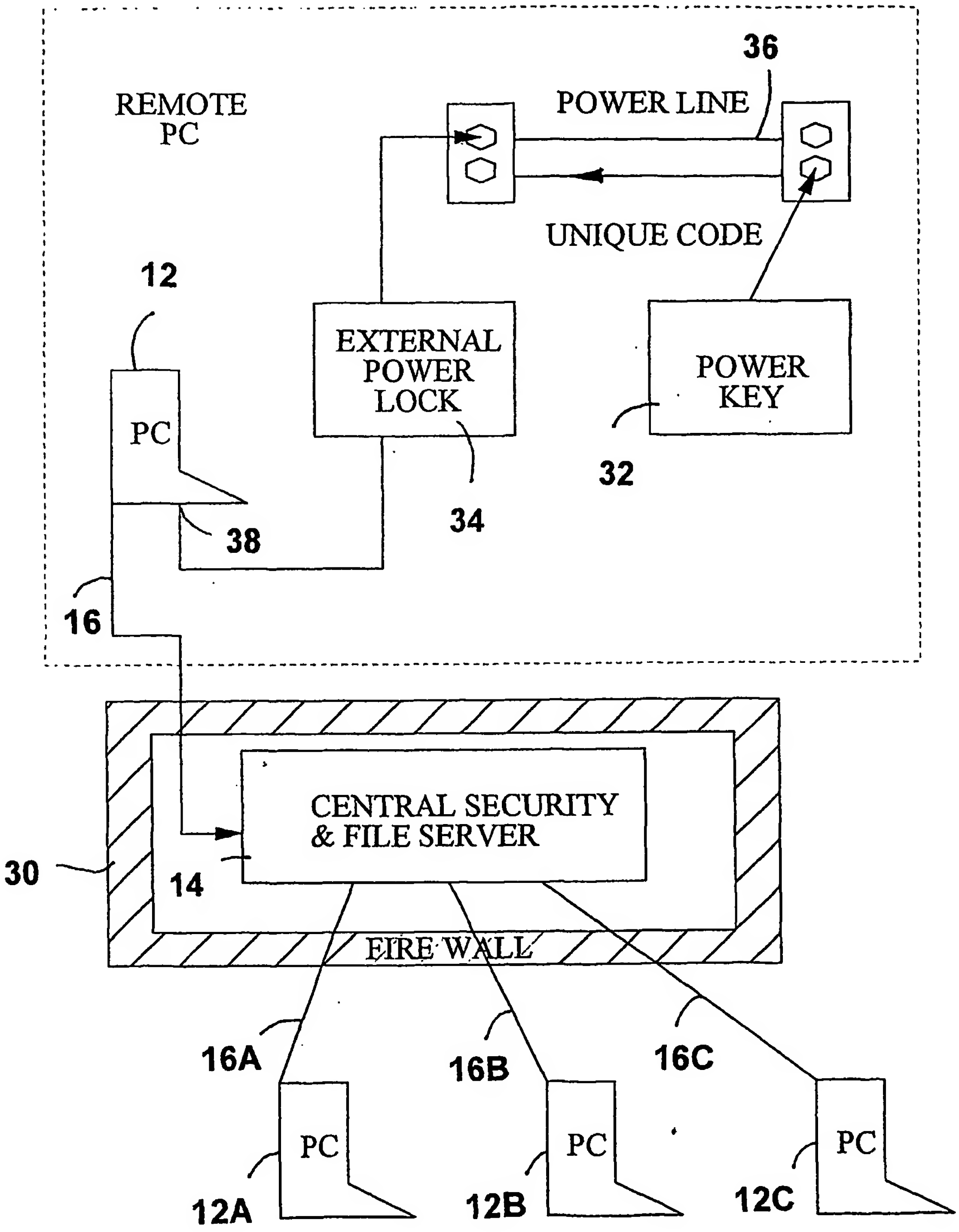

REMOTE PC
36
POWER LINE
UNIQUE CODE
12
PC
38
EXTERNAL POWER LOCK
34
POWER KEY
32
16
CENTRAL SECURITY & FILE SERVER
30
14
FIRE WALL
16A
16B
16C
PC
PC
PC
12A
12B
12C

Figure 12

INTERNATIONAL SEARCH REPORT

I national application No.
PCT/US01/09889

A. CLASSIFICATION OF SUBJECT MATTER

IPC(7) : G06F 1/26, 1/28, 1/30, 11/30, 12/14, 15/173; H04L 9/00, 9/32
US CL : 713/200, 201, 201, 300, 310, 300; 709/223, 224, 225
According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

U.S. : 713/200, 201, 201, 300, 310, 300; 709/223, 224, 225

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

Please See Extra Sheet.

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X --- Y	US 5,552,776 A (WADE ET AL) 03 SEPTEMBER 1996, COL. 3, LINES 18-34, COL. 4, LINE 53 THROUGH COL. 5, LINE 10, COL. 9, LINES 56-61	1,7-16,31 --- 2-6,17-30
Y	US 5,987,252 A (LEINO ET AL) 16 NOVEMBER 1999, COL. 11, LINES 29-38	2-6,17-30
Y	US 6,032,150 A (NGUYEN) 29 FEBRUARY 2000, SEE ABSTRACT, COL. 1, LINES 55-67, COL. 3, LINES 42-48	2-6,17-30

□ Further documents are listed in the continuation of Box C. □ See patent family annex.

*	Special categories of cited documents:	"T"	later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"A"	document defining the general state of the art which is not considered to be of particular relevance	"X"	document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"E"	earlier document published on or after the international filing date	"Y"	document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"L"	document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	"&"	document member of the same patent family
"O"	document referring to an oral disclosure, use, exhibition or other means		
"P"	document published prior to the international filing date but later than the priority date claimed		

Date of the actual completion of the international search

15 MAY 2001

Date of mailing of the international search report

22 JUN 2001

Name and mailing address of the ISA/US
Commissioner of Patents and Trademarks
Box PCT
Washington, D.C. 20231
Facsimile No. (703) 305-3230

Authorized officer Peggy Harrod
CHRISTOPHER A REVAK
Telephone No. (703) 305-9618

B. FIELDS SEARCHED

Electronic data bases consulted (Name of data base and where practicable terms used):

BRS (FILES: USPAT, EPO, JPO, DERWENT, IBM TDB'S), DIALOG (FILES: COMPSCI, ELECTRON, SOFTWARE)

search terms: power, powering, powered, start, starting, started, boot, booting, booted, up, on, password, credential, id, identifier, identification, transfer, transferred, transferring, transmit, transmission, transmitted, transmitting, file, data, information, software, application, encrypt, encrypted, encrypting, encryption, convert, converting, conversion, converted, server, control, controlled, controlling, manage, manages, management, managing, managed, workstation, pc, client, computer, terminal, applet, block, blocking, blocked, deny, denied, denying, allow, allowing, allowed, permit, permitting, permitted, accept, accepting, accepted